UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/756,863	01/10/2001	Toyohiko Yoshida	49657-921	6032

7590 06/16/2005

McDERMOTT, WILL & EMERY
600 13th Street, N.W.
Washington, DC 20005-3096

EXAMINER	
LI, AIMEE J	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 06/16/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/756,863	YOSHIDA, TOYOHIKO	
	Examiner	Art Unit	
	Aimee J. Li	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 May 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-12,14,16,18 and 19* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-12,14,16,18 and 19* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-12, 14, 16, and 18-19 have been considered.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: After Final Amendment as received on 13 May 2005.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1, 5, 12, 16 and 18-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hammond et al., U.S. Patent Number 5,638,525 (herein referred to as Hammond), in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (herein referred to as IBM).

5. Referring to claim 1, Hammond has taught in a processor operating with instructions in a first instruction architecture as a native instruction (Hammond column 4, lines 46-60 and Figure 2), an instruction translator used with an instruction memory to store an instruction in a second instruction architecture different from said first instruction architecture, for translating an instruction in said second instruction architecture into an instruction in said first instruction architecture for execution by said processor in said first instruction architectures (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6), said instruction translator comprising:

a. A translator for reading out an instruction from said instruction memory in response to a received first address in said instruction memory of an instruction to be executed by said processor and translating the read out instruction in said second instruction architecture into an instruction in said first instruction architecture, wherein said processor is configured to execute instructions only in said first instruction architecture (Hammond column 2, lines 32-60; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; Figure 2; and Figure 6);

b. An instruction cache for temporarily holding the instruction in said first instruction architecture after the translation by said translator in association with the first address in said instruction memory (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6).

6. Hammond has not explicitly taught a selector for searching said instruction cache in response to a received second address of an instruction to be executed by said processor, and for selectively outputting, based on a determination result of whether or not an instruction corresponding to the instruction of the second address is held in said instruction cache, an instruction output by said translator and the corresponding instruction held in said instruction cache. IBM has taught the selection of either an instruction stored in an instruction cache or an instruction from the main memory store based on if there is a cache miss (IBM p.1 lines 10-15 and p.2 lines 2-7) so as to reduce the delay associated with waiting for an instruction while a cache line is being retrieved on a cache miss (IBM p.1 lines 1-9). One of ordinary skill in the art

would have recognized that the instructions from main memory that are to be stored in the cache containing translated instructions have to come from the translator prior to being stored (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6). Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Hammond to include a cache bypass around the instruction cache of Hammond so that instructions can be selected from either the output of the translator or the instruction cache based on a cache hit/miss in order to reduce the delay associated with a fetch to memory on an instruction cache miss.

7. Regarding claim 5, Hammond has taught wherein said translator includes a translator which translates a plurality of instruction in said second instruction architecture read out from said instruction memory into one instruction in said first instruction architecture (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6).

8. While a plurality of translators is not explicitly taught, the translation of a plurality of instructions in the second instruction architecture into one instruction in the first instruction architecture is taught. The inclusion of a plurality of translators to perform the same function as a single translator provides no new or unexpected result over the prior art. Therefore, one of ordinary skill in the art would have found it obvious to duplicate the translator, creating a plurality of translators for translating instructions in a second instruction architecture into a single instruction in a first instruction architecture (see In re Harza, 274 F.2d 669, 671, 124 USPQ 378, 380 (CCPA 1960)).

9. Regarding claim 12, Emma has taught an instruction memory attached with a translator, used with a processor configured to execute instructions in a first instruction architecture as a native instruction (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6), comprising:

a. An instruction storage unit to store an instruction in a second instruction architecture (Hammond column 4, lines 46-60 and Figure 2),

b. An instruction translator to translate an instruction in said second instruction architecture output from said instruction storage unit into an instruction in said first instruction architecture for application to said processor, wherein said processor is configured to execute instructions only in said first instruction architecture (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6), said instruction translator including:

i. A translator to read out the instruction in said second instruction architecture from said instruction storage unit in response to a received first address of an instruction to be executed by said processor and translate the read out instruction in said second instruction architecture in the instruction in said first instruction architecture (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6),

ii. An instruction cache to temporarily hold the instruction in said first instruction architecture after the translation by said translator in

association with the first address (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6).

10. Hammond has not explicitly taught a selector to search said instruction cache in response to a received second address of an instruction to be executed by said processor and selectively output to said processor, based on a determination result of whether or no an instruction corresponding to the instruction of the address is held in said instruction cache, an instruction output by said translator or the corresponding instruction in said first instruction architecture held in said instruction cache. IBM has taught the selection of either an instruction stored in an instruction cache or an instruction from the main memory store based on if there is a cache miss (IBM p.1 lines 10-15 and p.2 lines 2-7) so as to reduce the delay associated with waiting for an instruction while a cache line is being retrieved on a cache miss (IBM p.1 lines 1-9). One of ordinary skill in the art would have recognized that the instructions from main memory that are to be stored in the cache containing translated instructions have to come from the translator prior to being stored (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6). Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Hammond to include a cache bypass around the instruction cache of Hammond so that instructions can be selected from either the output of the translator or the instruction cache based on a cache hit/miss in order to reduce the delay associated with a fetch to memory on an instruction cache miss.

11. Regarding claim 16, Emma has taught a data processing apparatus, comprising:

a. A processor operating with an instruction in a first instruction architecture as a native instruction (Hammond column 4, lines 46-60 and Figure 2),

a. A bus to which said processor is connected (Hammond Figure 5 and Figure 6 – busses are the connections between the elements),

b. A first instruction memory with a translator interconnected with said processor through said bus (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6),

c. A second instruction memory interconnected to said processor through said bus (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6),

d. Said first instruction memory with a translator including:

i. A first instruction storage unit to store an instruction in a second instruction architecture transferred from said processor through said bus (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6),

ii. An instruction translator to translate the instruction in said second instruction architecture output from said first instruction storage unit into an instruction in said first instruction architecture for application to said processor through said bus (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6),

e. Said second instruction memory including:

i. A second instruction storage unit to store an instruction in said first instruction architecture transferred from said processor through said bus (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6).

12. Hammond has not explicitly taught an instruction reading circuit responsive to an address signal applied form said processor through said bus for applying an instruction in said first instruction architecture output form said second instruction storage unit to said processor through said bus. IBM has taught the selection of either an instruction stored in an instruction cache or an instruction from the main memory store based on if there is a cache miss (IBM p.1 lines 10-15 and p.2 lines 2-7) so as to reduce the delay associated with waiting for an instruction while a cache line is being retrieved on a cache miss (IBM p.1 lines 1-9). One of ordinary skill in the art would have recognized that the instructions from main memory that are to be stored in the cache containing translated instructions have to come from the translator prior to being stored (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6). Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Hammond to include a cache bypass around the instruction cache of Hammond so that instructions can be selected from either the output of the translator or the instruction cache based on a cache hit/miss in order to reduce the delay associated with a fetch to memory on an instruction cache miss.

13. Regarding claim 18, Hammond in view of IBM has taught wherein the number of clock cycles spent by said processor to access said first instruction memory through said bus is larger

than the number of clock cycles spent by said processor to access said second instruction memory through said bus (IBM p.1 lines 1-9 and p.2 lines 2-7).

14. Regarding claim 19, Hammond has taught a second instruction memory (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6) with a translator interconnected to said processor through said bus, said second instruction memory with a translator including:

a. An instruction storage unit to store an instruction in a second instruction architecture different from said second instruction architecture, said instruction in said third instruction architecture being transferred from said processor through said bus (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6),

b. An instruction translation circuit responsive to an address signal applied from said processor through said bus, for translating an instruction in said third instruction architecture output from said instruction storage unit into an instruction in said first instruction architecture for application to said processor (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6).

15. Hammond has not explicitly taught a third instruction memory with a translator for translating a third instruction architecture into a first instruction architecture is not explicitly taught. However, adding a third instruction memory and corresponding third instruction architecture provides no new or unexpected result over the prior art except for allowing the additional translation of a third instruction architecture. But this could be achieved simply by

replacing the second instruction architecture with the third architecture, as there are no claim limitations that require the two instruction architectures to be translated simultaneously. Therefore, one of ordinary skill in the art would have found it obvious to duplicate the hardware used to operate on a second instruction architecture as taught by Emma in view of IBM in order to operate on a third instruction architecture (see In re Harza, 274 F.2d 669, 671, 124 USPQ 378, 380 (CCPA 1960)).

16. Claims 2-3 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hammond et al., U.S. Patent Number 5,638,525 (herein referred to as Hammond), in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (herein referred to as IBM), as applied to claim 1 above, and further in view of Dickol et al., U.S. Patent No. 5,875,336 (herein referred to as Dickol).

17. Regarding claim 2, Hammond has taught wherein said second instruction architecture is a variable length instruction architecture (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6). In regards to Hammond, CISC architecture is a variable length architecture. Hammond has not taught said translator includes a variable length translator for translating an instruction in said second instruction architecture read out from said instruction memory into one or more instruction in said first instruction architecture, the number of which depends on an instruction length of the read out instruction in said second instruction architecture. Dickol has taught a variable length instruction architecture being translated in real time into one or more instructions of a second instruction architecture based on the varying lengths of instructions in the variable length instruction architecture (Dickol, Col.3 lines 29-38 and Col.4 lines 13-14, 21-27) in order

to improve performance when executing non-native instructions on more common native-instruction-based hardware (Dickol, Col.1 lines 47-61 and Col.2 lines 15-18). One of ordinary skill in the art would have recognized that a priority of microprocessor designers is to improve performance while minimizing cost and hardware complexity. Therefore, one of ordinary skill in the art at the time of the invention would have found it obvious to modify Emma in view of IBM to further decode instructions from a non-native variable length instruction set into one or more native instructions in order to improve processor performance when executing non-native instructions while minimizing the additional hardware required to do so.

18. Regarding claim 3, Emma in view of IBM in further view of Dickol has taught wherein said variable length translator translates the instruction in said second instruction architecture read out from said instruction memory into an instruction in said first instruction architecture having a total length depending on and larger than the instruction length of said read out instruction in said second instruction architecture (Dickol, Col.3 lines 29-38 and Col.4 lines 13-14, 21-27).

19. Claim 4 is rejected under 35 U.S.C. 103(a) as being unpatentable over Hammond et al., U.S. Patent Number 5,638,525 (herein referred to as Hammond), in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (herein referred to as IBM), as applied to claim 1 above, in view of Dickol et al., U.S. Patent No. 5,875,336 (herein referred to as Dickol), and further in view of Goettelmann et al., U.S. Patent No. 5,313,614.

20. Regarding claim 4, Hammond in view of IBM in view of Dickol have not explicitly taught wherein each instruction in said first instruction architecture includes one or a plurality of sub instructions, and the number of the sub instructions included in the instruction in the first

instruction architecture translated by said variable length translator depends on the instruction length of said read out instruction in said second instruction architecture. Goettelmann has taught the translation from a source instruction architecture (Goettelmann, "source machine code" of Fig.11) into a native instruction architecture (Goettelmann "translated code" of Fig.11), wherein the native architecture contains a plurality of sub-instructions (Goettelmann, "expanded intermediate language code" of Fig.11), sub-instructions which can then be removed if necessary in order to reduce the translated code size (Goettelmann, Col.4 lines 6-18). One of ordinary skill in the art would have recognized that it is desirable to reduce the size of instruction code so that less hardware (memory space) is needed, thus lowering costs. Therefore, one of ordinary skill in the art would have found it obvious to modify the instruction translator of Emma in view of IBM in view of Dickol to further translate instructions from a source architecture into a native architecture that includes a plurality of sub-instructions in order to reduce the amount of instruction storage required to store the translated code. Furthermore, because the number of native instructions are dependent on the length of the varying-length source instruction as shown above, and because the number of sub instructions depend on the native instructions, then the number of sub-instructions depend on the length of the source instructions.

21. Claims 6-7 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hammond et al., U.S. Patent Number 5,638,525 (herein referred to as Hammond), in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (herein referred to as IBM), as applied to claim 1 above, as applied to claim 1 above, and further in view of Goettelmann et al., U.S. Patent No. 5,313,614 (herein referred to as Goettelmann).

22. Regarding claim 6, Hammond in view of IBM have not explicitly taught wherein:

a. Each instruction in said first instruction architecture can include one or a plurality of sub instructions,

b. Said translator translates a plurality of instructions in said second instruction architecture read out from said instruction memory into an instruction in said first instruction architecture including sub instructions, the number of which depends on the number of said plurality of instructions.

23. Goettelmann has taught the translation from a source instruction architecture (Goettelmann, "source machine code" of Fig.11) into a native instruction architecture (Goettelmann, "translated code" of Fig.11), wherein the native architecture contains a plurality of sub-instructions (Goettelmann, "expanded intermediate language code" of Fig.11), sub-instructions which can then be removed if necessary in order to reduce the translated code size (Goettelmann, Col.4 lines 6-18). One of ordinary skill in the art would have recognized that it is desirable to reduce the size of instruction code so that less hardware (memory space) is needed, thus lowering costs. Therefore, one of ordinary skill in the art would have found it obvious to modify the instruction translator of Emma in view of IBM to further translate instructions from a source architecture into a native architecture that includes a plurality of sub-instructions in order to reduce the amount of instruction storage required to store the translated code. Furthermore, because the number of native instructions are dependent on the length of the varying-length source instruction as shown above, and because the number of sub instructions depend on the native instructions, then the number of sub-instructions depend on the length of the source instructions.

24. Regarding claim 7, Hammond in view of IBM in further view of Goettelmann have taught wherein the number of sub instructions included in the instruction in said first instruction architecture after said translation is equal to the number of said plurality of instructions (see Goettelmann "BEQ Label instruction" of "source machine code", its corresponding "BFALSE FlagZ, Label" instruction of "translated code", with associated sub-instruction "BFALS.d FlagZ, Label" of "expanded intermediate language code", all of Fig.11).

25. Claims 8 and 10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hammond et al., U.S. Patent Number 5,638,525 (herein referred to as Hammond), in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (herein referred to as IBM), further in view of Gregor, U.S. Patent No. 5,023, 776 (herein referred to as Gregor).

26. Regarding claim 8, Hammond has taught wherein said translator translates said read out instruction in said second instruction architecture into one or a plurality of instructions as the instruction in said first instruction architecture (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6). Hammond in view of IBM has not explicitly taught where said instruction translator further comprises a controller for controlling said instruction cache so that said instruction cache holds each of said one or said plurality of instruction held in said instruction cache as an entry which can be invalidated in one of first and second conditions. Gregor has taught the holding of a cache line within a cache so that it cannot be replaced until an EOP signal is detected in order to reduce cache busy time and improve processor performance (Gregor, Col.22 lines 36-67). One of ordinary skill in the art would have recognized that it is desirable, as well as a main goal of

microprocessor design, to improve the performance of a microprocessor. Therefore, one of ordinary skill in the art would have found it obvious to modify the cache of Emma in view of IBM to hold a plurality of instructions in a cache until an EOP signal is detected. Furthermore, because the claim language is in the alternative format, only one of the two invalidation condition requirements is required to be met, and thus the EOP signal that is detected (Gregor, Col.22 lines 56-57) can be considered such a signal.

27. Regarding claim 10, Hammond in view IBM in further view of Gregor has taught wherein said controller outputs a signal asserted when a new instruction cannot be held in said instruction cache without invalidating an entry which can be invalidated in the second condition (Gregor, Col.22 lines 36-67).

28. Claims 9 and 11 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hammond et al., U.S. Patent Number 5,638,525 (herein referred to as Hammond), in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (herein referred to as IBM), in view of Gregor, U.S. Patent No. 5,023, 776 (herein referred to as Gregor), in further view of Schacham et al., U.K. Patent Application GB220481A.

29. Regarding claim 9, Hammond in view of IBM in view of Gregor have taught wherein said first condition is a holding control condition by hardware control based on a prescribed algorithm by said instruction cache (Gregor, Col.22 lines 36-67). Hammond in view of IBM in further view of Gregor has not explicitly taught wherein the said second condition is a condition in which an explicit invalidation instruction is applied from the outside of said instruction cache. However, in the parent claim of claim 9 only one of the two invalidation conditions must be met. Therefore, because Hammond in view of IBM in further view of Gregor has satisfied one of the

conditions, namely the holding control condition as shown above, the claim language is satisfied. Furthermore, even though the alternative form claim language is met, Schacham has taught that a cache invalidation instruction can be executed to invalidate the entire instruction cache or a portion of it (Schacham, p.4 lines 28-30) in order to maintain proper cache coherence and provide better processor performance (Schacham, p.2 lines 8-23). One of ordinary skill in the art would have recognized that it is desirable, as well as a main goal of microprocessor design, to improve the performance of a microprocessor. Therefore, one of ordinary skill in the art would have found it obvious to modify the instruction cache of Emma in view of IBM in view of Gregor to allow a cache invalidation instruction to be executed so that cache coherency is maintained and processor performance is improved.

30. Regarding claim 11, Hammond has taught wherein said translator translates the read out instruction in said second instruction architecture into the plurality of instructions as the instruction in said first instruction architecture (Hammond column 2, lines 32-52; column 14, lines 55-59; column 15, lines 13-27; column 16, lines 17-19 and 37-45; Figure 5; and Figure 6). Hammond in view of IBM in view of Gregor have not explicitly taught where said controller provides said first condition with one of said plurality of instructions and said second condition with each of said plurality of instructions but said one instruction. Schacham has taught that a cache invalidation instruction can be executed to invalidate the entire instruction cache or a portion of it (Schacham, p.4 lines 28-30) in order to maintain proper cache coherence and provide better processor performance (Schacham, p.2 lines 8-23). One of ordinary skill in the art would have recognized that it is desirable, as well as a main goal of microprocessor design, to improve the performance of a microprocessor. Therefore, one of ordinary skill in the art would

have found it obvious to modify the instruction cache of Emma in view of IBM in view of Gregor to allow a cache invalidation instruction to be executed so that cache coherency is maintained and processor performance is improved. Furthermore, Emma in view of IBM in further view of Gregor in view of Schacham has taught the invalidation of instructions in a cache based upon one of two conditions, namely a holding control condition as taught by Gregor, and a cache invalidation instruction as taught by Schacham. One of ordinary skill in the art would have recognized that because some instructions will have the first condition associated with them, and others will have the second condition associated with them. Therefore, one of ordinary skill in the art would have found it obvious that the processor of Emma in view of IBM in further view of Gregor in view of Schacham will provide one instruction with a first condition, and other instructions with a second condition.

31. Claim 14 is rejected under 35 U.S.C. 103(a) as being unpatentable over Hammond et al., U.S. Patent Number 5,638,525 (herein referred to as Hammond), in view of IBM Technical Disclosure Bulletin, *Instruction Cache Bypass During Cache Reload* (herein referred to as IBM), in view of Mallick, U.S. Patent Number 5,953,520 (herein referred to as Mallick). Hammond in view of IBM has not taught an address translator to translate an address at the time of reading from said instruction storage unit. Mallick has taught an address translator to translate an address at the time of reading from said instruction storage unit (Mallick Abstract; column 3, lines 25-30; and column 12, lines 36-55). A person of ordinary skill in the art at the time the invention was made would have recognized that incorporating the address translator of Mallick allows memory addressed in the second architecture to be accessed by memory locations specified in the first architecture (Mallick column 2, lines 62-64), thereby improving

compatibility between the two architectures. Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the address translator of Mallick in the device of Hammond in view of OBM to improve compatibility between the two instruction architectures.

Response to Arguments

32. Applicant's arguments, see After Final Amendment, filed 13 May 2005, with respect to the rejection(s)of claim(s) 1-12, 14, 16, and 18-19 have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of the above.

Conclusion

33. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

a. Slavenburg, U.S. Patent Number 4,750,108, has taught translating macro-instructions into micro-instructions.

b. Blomgren et al., U.S. Patent Number 5,781,750, has taught translating and/or emulating CISC instructions into RISC instructions.

c. Theogarajan et al., U.S. Patent Number 6,308,257, has taught translating variable length instructions into fixed length instructions.

d. IBM Technical Disclosure Bulletin, published 01 March 1994, has taught translating CISC instructions to RISC instructions for execution in RISC hardware.

34. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Aimee J. Li whose telephone number is (571) 272-4169. The examiner can normally be reached on M-T 7:30am-5:00pm.

35. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

36. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

AJL
Aimee J. Li
27 May 2005

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100